A Variable Structure Control Approach to Active Queue Management for TCP With ECN

Peng Yan, *Member, IEEE*, Yuan Gao, and Hitay Özbay, *Senior Member, IEEE*

***Abstract*—It has been shown that the transmission control protocol (TCP) connections through the congested routers can be modeled as a feedback dynamic system. In this paper, we design a variable structure (VS) based control scheme in active queue management (AQM) supporting explicit congestion notification (ECN). By analyzing the robustness and performance of the control scheme for the nonlinear TCP/AQM model, we show that the proposed design has good performance and robustness with respect to the uncertainties of the round-trip time (RTT) and the number of active TCP sessions, which are central to the notion of AQM. Implementation issues are discussed and *ns* simulations are provided to validate the design and compare its performance to other peer schemes' in different scenarios. The results show that the proposed design significantly outperforms the peer AQM schemes in terms of packet loss ratio, throughput and buffer fluctuation.**

***Index Terms*—Active queue management (AQM), explicit congestion notification (ECN), robustness, transmission control protocol (TCP)/Internet protocol (IP), variable structure (VS) control.**

I. Introduction

TRAFFIC congestion on the Internet is one of the major communication problems experienced by millions of users. Since the landmark work on the original congestion avoidance scheme in transmission control protocol (TCP) [12], many efforts have been devoted to improve the Internet congestion control performance. Earliest efforts are focused on TCP enhancement. Recently, for the purpose of alleviating congestion problem for Internet protocol (IP) networks and providing some notion of quality of service (QoS), active queue management (AQM) techniques have been proposed [8]–[10], [17], [19], [22], [23]. Current most widely-used queue management scheme, DropTail has been observed to have some drawbacks, namely, link underutilization, queue fluctuation and global synchronization of competing TCP connections, due to its heuristic and empirical nature. Meanwhile, randon early detection (RED) is an AQM mechanism proposed in [8] to reduce link congestion and global synchronization by earlier congestion notification. Unfortunately, it has been shown that the performance of a RED router is very sensitive to link's traffic load and its parameter setting, and it is hard to reduce the queue fluctuation by only adjusting RED's parameters [3]. The drawbacks of RED prompt researchers to propose modifications and alternatives, such as BLUE [6], stabilized RED (SRED) [19], and fai RED (FRED) [16].

Notably, the TCP connections through the congested routers can be modeled as a feedback dynamic system, where control theory-based approaches can be used to analyze the network behavior, tune AQM's parameter settings, and design new AQM schemes. We refer to [13], [17], and [18] for the details of the nonlinear dynamic models for the TCP's additive increase and multiplicative decrease (AIMD) behavior and the AQM's congestion feedback. It is believed that the control system-based analysis offers new insight into the AQM design. In [9], a control theoretic analysis was given for RED, which provided a more systematic and in-depth study on RED parameter tuning; and [10] developed a PI controller as a new AQM scheme using linear system analysis. On the other hand, the AQM congestion control can be viewed as a convex optimization problem for the dynamic system [13], [14], [17], where steady-state properties and equilibriums are investigated. Meanwhile, explicit congestion notification (ECN) [22] has been proposed, where packet marking is used as congestion indication. We refer to the AVQ algorithm [14] for an ECN-based AQM design. Taking the stability and performance indices into consideration, the previous control theoretic AQM schemes achieve better performance in terms of high link utilization and low packet loss ratio.

Due to the challenging nature of nonlinearity in the TCP dynamics, most of the current results on the AQM analysis and design are based on linearized models which are valid only in the neighborhood of the equilibrium points [9], [10], [14], [23]. To further complicate the situation, the TCP/AQM dynamics have time varying round-trip times (RTTs) and uncertainties with respect to the number of active TCP sessions through the congested AQM router, which requires more robustness for the designed schemes. In this paper, we introduce a robust variable structure (VS) based AQM scheme designed directly for the nonlinear TCP model, which to the best of our knowledge has not yet been employed to TCP/AQM system analysis and design. The motivations behind this work are: 1) VS sliding mode control is robust and powerful for nonlinear systems, thus, being well-suited for an AQM scheme and 2) DropTail, a widely deployed AQM scheme, can be treated as a VS controller in the sense that its dropping policy is $p(t) = 1$ when $q(t) > q_{\max}$, and $p(t) = 0$ when $q(t) \leq q_{\max}$, where $p(t)$ is the dropping probability, $q(t)$ and $q_{\max}$ are the occupied buffer and the maximum buffer length, respectively. Note that the undesirable behavior

Manuscript received February 23, 2003; revised February 3, 2004. Manuscript received in final form June 14, 2004. Recommended by Associate Editor C. C. Lee. This work was supported by the National Science Fundation under Grant ANI-0073725.

P. Yan is with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210 USA (e-mail: yanp@ece.osu.edu).

Y. Gao is with the Network Planning and Standards Center, Bell Laboratories, Lucent Technologies, Holmdel, NJ 07733 USA (e-mail: yuangao@lucent.com).

H. Özbay is with Department of Electrical and Electronics Engineering, Bilkent University, Ankara 06800, Turkey (e-mail: hitay@bilkent.edu.tr). He is on leave from the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210 USA.

Digital Object Identifier 10.1109/TCST.2004.839566

of DropTail (e.g., heavy queue fluctuation and high dropping rate), from the VS control perspective, is due to the fact that its binomial control does not fit TCP dynamics to enforce a stable sliding mode.

In this study, we present guidelines to design VS controller for AQM supporting ECN. By analyzing the robustness and performance of the proposed AQM scheme, we show that the proposed design has good asymptotic properties as well as stability robustness with respect to RTTs and the number of the active TCP sessions through the router, which are central to the notion of AQM. Meanwhile, we also discuss the implementation issues and compare our design against related works. Our VS-based AQM scheme is shown via *ns* simulations to be a robust controller that performs better than a number of well-known AQM schemes under different scenarios.

This paper is organized as follows. In Section II, we discuss the nonlinear TCP traffic dynamics and develop a VS-based AQM controller supporting ECN. The robustness and performance of the proposed AQM scheme are analyzed in Section III, where implementation issues are also discussed. Then we summarize the related AQM schemes and provide comparisons in Section IV. The *ns* simulation results are presented in Section V to validate our design, followed by the concluding remarks in Section VI.

II. Variable Structure Control in AQM

A. Nonlinear TCP Dynamics

In [13], a nonlinear dynamic model for TCP congestion control was derived, where the network topology was assumed to be a single bottleneck with M homogeneous TCP sources that share the bottleneck link and have roughly the same RTTs, but do not necessarily transverse the same path. For TCP with ECN, the AIMD behavior in congestion avoidance phase can be modeled as follows: each positive acknowledgment increases the value of congestion window cwnd by $1/\mathrm{cwnd}$ while each congestion indication (ECN) reduces the cwnd by half, thus, the expected change in congestion window is given by $(1-p)/\mathrm{cwnd} - (\mathrm{cwnd}/2)p$, where p is the marking probability. Aggregating the M TCP flows through one congested router, we have the expected change in the congestion window cwnd per update step

$$\Delta_{\mathrm{cwnd}} = \frac{1}{M}\left(\frac{1-p(t)}{\mathrm{cwnd}} - \frac{\mathrm{cwnd}}{2}p(t)\right) \tag{1}$$

where $0 \leq p(t) \leq 1$ is the marking probability and M is the number of the active TCP sessions. Denote $r(t)$ to be the incoming traffic rate per unit time and $R(t)$ the round trip time delay, we have $r(t) = M\mathrm{cwnd}/R(t)$. We assume the variation of $R(t)$ is much slower than $r(t)$. Since the time between update steps is about $\Delta R := R(t)/(M\mathrm{cwnd})$, the expected change in the aggregated rate $r(t)$ per unit time is approximately

$$\frac{\Delta r}{\Delta R} = \frac{M\Delta_{\mathrm{cwnd}}/R(t)}{R(t)/(M\mathrm{cwnd})} = \frac{M}{R^2(t)} - \left(\frac{M}{R^2(t)} + \frac{r^2(t)}{2M}\right)p(t). \tag{2}$$

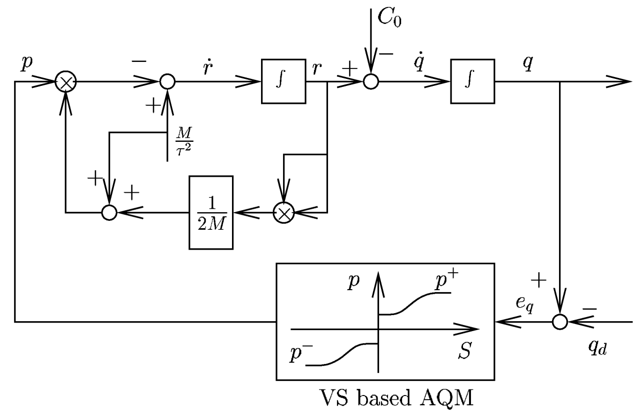

Fig. 1. Aggregated dynamics of TCP and VS-based AQM.

Motivated by this calculation, we have

$$\begin{cases} \dot{r}(t) = \frac{M}{R^2(t)} - \left(\frac{M}{R^2(t)} + \frac{r^2(t)}{2M}\right)p(t) \\ \dot{q}(t) = r(t) - C_0, \end{cases} \tag{3}$$

and

$$R(t) = \frac{q(t)}{C_0} + T_p \tag{4}$$

where $q(t)$ is the instantaneous queue length on the router, T_p is the propagation delay and C_0 is the link capacity. Note that (3) is a simplified model without considering the time delays in $r(t)$ and $p(t)$. We will show in Section V that this simplification results in the steady-state error of the queue size with VS AQM controllers.

We will replace $R(t)$ with $\tau(t)$ in (3), assuming an unknown continuous function $\tau(t)$ satisfying

$$T_p \leq T_0 \leq \tau(t) \leq T_1 \leq \frac{q_{\max}}{C_0} + T_p \tag{5}$$

where $q_{\max}$ is the buffer length of the router. Note that (5) captures the time varying nature of $R(t)$. But it introduces conservatism to the analysis in the sense that $\tau(t)$ is independent of $q(t)$. We further assume the number of active TCP connections to be uncertain, obeying

$$0 < M^- \leq M \leq M^+ \tag{6}$$

which is more reasonable in practice.

Recall that most of the current AQM schemes [9], [10], [14], [23] are based on the linearized model, which are valid only in the neighborhood of the operating points. Meanwhile, some important parts of TCP are even not included in the nonlinear model (3): 1) nonresponsive UDP flows are not modeled and 2) the impact of short-lived connections (the so called *web mice*), such as Telnet and HTTP. Taking the nonlinearity and the unmodeled uncertainty into consideration, we believe that variable structure sliding model control would be an ideal methodology for a robust AQM.

B. VS-Based AQM with ECN

The nonlinear TCP dynamics and the VS-based AQM can be modeled as a feedback control system depicted in Fig. 1, where the VS controller uses the queue and traffic incoming rate information to generate marking rate as congestion indication.

Let $e_q = q(t) - q_d$ and denote $x_1 = e_q, x_2 = \dot{e}_q$, where q_d is the desired queue length. We have

$$x_2 = \dot{e}_q = \dot{q}(t) = r(t) - C_0 \tag{7}$$

and the plant (3) can be described as

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{M}{\tau^2(t)} - \left(\frac{M}{\tau^2(t)} + \frac{(x_2+C_0)^2}{2M}\right) p(t) \end{cases}. \tag{8}$$

Note that (8) is an affine nonlinear system in the form of

$$\dot{x} = f(x,t) + B(x,t)p(t)$$

where

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad f(x,t) = \begin{bmatrix} x_2 \\ \frac{M}{\tau^2(t)} \end{bmatrix}$$

and

$$B(x,t) = \begin{bmatrix} 0 \\ -\left(\frac{M}{\tau^2(t)} + \frac{(x_2+C_0)^2}{2M}\right) \end{bmatrix}.$$

Thus, the equivalent control method (ECM) can be used to construct the VS control law. We select the sliding mode surface as

$$S(x,t) = k_S x_1 + x_2, \quad k_S > 0 \tag{9}$$

which corresponds to a linear combination of the queue length error and the error between incoming traffic rate and link capacity.

The corresponding existence condition [25] for the sliding mode is

$$D = -\frac{GB + (GB)^T}{2} > 0 \tag{10}$$

where $G := \partial S/\partial x$. Observe

$$b(x_2) := \frac{M}{\tau^2(t)} + \frac{(x_2+C_0)^2}{2M} > 0$$

and

$$GB = \begin{bmatrix} \frac{\partial S}{\partial x_1} & \frac{\partial S}{\partial x_2} \end{bmatrix} \begin{bmatrix} 0 \\ -b(x_2) \end{bmatrix} = -b(x_2) < 0.$$

It is straightforward that (10) is satisfied. The typical sliding mode controller [25] is given by

$$p = \bar{\alpha} F(x,t) \mathrm{sign}(S), \quad \bar{\alpha} > 1 \tag{11}$$

where

$$p_{\mathrm{eq}} = -(GB)^{-1} G f = \frac{k_S x_2 + \frac{M}{\tau^2(t)}}{b(x_2)} \tag{12}$$

and $F(x,t)$ is any continuous function satisfying $F(x,t) > |p_{\mathrm{eq}}(x,t)|$.

Unfortunately, controller (11) is not practically feasible because the negative part of $p(t)$ in (11) is out of the bound of $0 \le p(t) \le 1$. In what follows, we would like to improve the sliding mode controller by introducing a feedback term. Define

$$\bar{b}(x_2) = \frac{\bar{M}}{\bar{\tau}^2} + \frac{(x_2+C_0)^2}{2\bar{M}} \tag{13}$$

where $\bar{M} := (M^- + M^+)/2$ and $\bar{\tau} := (T_0 + T_1)/2$ are the nominal values of M and $\tau(t)$, respectively. According to the sliding mode existence condition and the robustness criteria, we construct the VS controller as following:

$$\begin{aligned} p(x,t) &= \frac{\bar{M}}{\bar{\tau}^2 \bar{b}(x_2)} + \left(\alpha \frac{T_1}{\sqrt{2}} k_S \left|\frac{x_2}{r(t)}\right| + \delta\right) \mathrm{sign}(S(x,t)) \\ &= \begin{cases} p^+(x,t) & \text{if } S(x,t) > 0 \\ p^-(x,t) & \text{if } S(x,t) < 0 \end{cases} \end{aligned} \tag{14}$$

where $\alpha \ge 1, \delta > 0$ are constants and

$$\begin{aligned} p^+(x,t) &= \frac{\bar{M}}{\bar{\tau}^2 \bar{b}(x_2)} + \left(\alpha \frac{T_1}{\sqrt{2}} k_S \left|\frac{x_2}{x_2+C_0}\right| + \delta\right) \\ p^-(x,t) &= \frac{\bar{M}}{\bar{\tau}^2 \bar{b}(x_2)} - \left(\alpha \frac{T_1}{\sqrt{2}} k_S \left|\frac{x_2}{x_2+C_0}\right| + \delta\right). \end{aligned} \tag{15}$$

Stability and robustness with respect to the proposed VS controller will be discussed in the next section, where we also provide guidelines of choosing the parameters k_S, α, and δ.

III. Robustness and Performance Analysis

In order to guarantee the robust stability of the closed-loop system, the existence condition for the sliding mode should be satisfied, which is given by the following theorem.

Theorem 1: The VS controller (14) robustly stabilizes the nonlinear system (8) for all $\tau(t)$ and M obeying (5) and (6), respectively, if $\alpha \ge 1$ and

$$\delta > \gamma := \frac{1}{4}\left(\frac{\bar{M}}{M^-} + \frac{\bar{\tau}}{T_0}\right)\left(\frac{M^+ - M^-}{M^+ + M^-} + \frac{T_1 - T_0}{T_1 + T_0}\right). \tag{16}$$

Proof: First, notice that

$$b(x_2) = \frac{M}{\tau^2(t)} + \frac{(x_2+C_0)^2}{2M} \ge \frac{\sqrt{2}}{\tau(t)}|x_2 + C_0|. \tag{17}$$

Similarly, we have $\bar{b}(x_2) \ge (\sqrt{2}/\bar{\tau})|x_2 + C_0|$. Thus

$$\begin{aligned} &\left|\frac{\bar{M}}{\bar{\tau}^2 \bar{b}(x_2)} - \frac{M}{\tau^2(t) b(x_2)}\right| \\ &= \left|\frac{\left(\frac{\bar{M}\tau^2}{2M} - \frac{M\bar{\tau}^2}{2\bar{M}}\right)}{\bar{\tau}^2 \tau^2 \bar{b}(x_2) b(x_2)}\right| (x_2+C_0)^2 \\ &\le \left|\frac{\left(\frac{\bar{M}\tau^2}{2M} - \frac{M\bar{\tau}^2}{2\bar{M}}\right)}{2\tau\bar{\tau}}\right| \\ &= \left|\frac{(\bar{M}\tau)^2 - (M\bar{\tau})^2}{4\tau\bar{\tau}M\bar{M}}\right| \\ &= \left|\frac{\bar{M}\tau + M\bar{\tau}}{4M\tau}\right| \left|\frac{(\bar{M}\tau - \bar{M}\bar{\tau}) + (\bar{M}\bar{\tau} - M\bar{\tau})}{\bar{M}\bar{\tau}}\right| \\ &\le \gamma \end{aligned} \tag{18}$$

which is straightforward from (5), (6), (16), and (17).

Recall (9), the time derivative of $S(x,t)$ along the trajectory of (8) under the control (14) is given as

$$\begin{aligned}\dot{S}(x,t) &= k_S x_2 + \frac{M}{\tau^2(t)} - b(x_2)p(x,t)\\ &= k_S x_2 + \frac{M}{\tau^2(t)} - b(x_2)\left[\frac{\bar{M}}{\bar{\tau}^2(t)\bar{b}(x_2)}\right.\\ &\quad \left. + \left(\alpha\frac{T_1}{\sqrt{2}}k_S|x_2/r(t)| + \delta\right)\operatorname{sign}(S(x,t))\right]. \quad (19)\end{aligned}$$

Note that $b(x_2)$ and $\bar{b}(x_2)$ are positive. Invoke (16), (17), and (18), we have the following.

1) If $S(x,t) > 0$

$$\begin{aligned}\dot{S} &= k_S x_2 - \frac{T_1 b(x_2)}{\sqrt{2}}k_S|x_2/r(t)| + \frac{M}{\tau^2} - \frac{b(x_2)\bar{M}}{\bar{\tau}^2\bar{b}(x_2)}\\ &\quad - b(x_2)\left((\alpha-1)\frac{T_1 k_S}{\sqrt{2}}|x_2/r(t)| + \delta\right)\\ &< k_S x_2 - \frac{T_1 b(x_2)}{\sqrt{2}}k_S|x_2/r(t)| + \frac{M}{\tau^2} - \frac{b(x_2)\bar{M}}{\bar{\tau}^2\bar{b}(x_2)}\\ &\quad - \gamma b(x_2)\\ &\le k_S|x_2|\left(1 - \frac{T_1 b(x_2)}{\sqrt{2}|x_2 + C_0|}\right)\\ &\quad + b(x_2)\left(\left|\frac{M}{\tau^2 b(x_2)} - \frac{\bar{M}}{\bar{\tau}^2\bar{b}(x_2)}\right| - \gamma\right)\\ &\le k_S|x_2|\left(1 - \frac{\tau(t)b(x_2)}{\sqrt{2}|x_2 + C_0|}\right) \le 0. \quad (20)\end{aligned}$$

2) If $S(x,t) < 0$

$$\begin{aligned}\dot{S} &= k_S x_2 + \frac{T_1 b(x_2)}{\sqrt{2}}k_S|x_2/r(t)| + \frac{M}{\tau^2} - \frac{b(x_2)\bar{M}}{\bar{\tau}^2\bar{b}(x_2)}\\ &\quad + b(x_2)\left((\alpha-1)\frac{T_1 k_S}{\sqrt{2}}|x_2/r(t)| + \delta\right)\\ &> k_S x_2 + \frac{T_1 b(x_2)}{\sqrt{2}}k_S|x_2/r(t)| + \frac{M}{\tau^2} - \frac{b(x_2)\bar{M}}{\bar{\tau}^2\bar{b}(x_2)}\\ &\quad + \gamma b(x_2)\\ &\ge k_S|x_2|\left(\frac{T_1 b(x_2)}{\sqrt{2}|x_2 + C_0|} - 1\right)\\ &\quad + b(x_2)\left(\gamma - \left|\frac{M}{\tau^2 b(x_2)} - \frac{\bar{M}}{\bar{\tau}^2\bar{b}(x_2)}\right|\right)\\ &\ge k_S|x_2|\left(\frac{\tau(t)b(x_2)}{\sqrt{2}|x_2 + C_0|} - 1\right) \ge 0 \quad (21)\end{aligned}$$

which indicates $\lim_{S\to 0} S(x,t)\dot{S}(x,t) < 0$. According to the existence condition of the sliding mode [25], the proposed VS controller is robustly stable and the state trajectory of the feedback system converges to the sliding mode. Thus, the proof is complete. ■

Inserting the equivalent controller (12) into system (8), we obtain the motion on the sliding manifold $S(x,t) = 0$

$$\dot{x} = -k_S x. \quad (22)$$

Thus, the instantaneous queue length $q(t)$ exponentially converges to the desired queue length q_d with decay rate k_S once the trajectory reaches the sliding mode surface.

Remark 1: Note that k_S is the decay rate of $q(t) - q_d$. For the purpose of fast convergence, we would like to pick up k_S as large as possible. On the other hand, a larger k_S results in a larger magnitude of the marking probability $p(t)$ (from (14) and (15)), which increases the risk of saturation ($0 \le p(t) \le 1$). Practically, we choose k_S so that the two terms $k_S x_1$ and x_2 in (9) are balanced (e.g., $k_S = 10$ is feasible when $q_d = 100$ and $C_0 = 1000$).

Another important aspect of the system performance is how fast the closed-loop system reaches the sliding mode surface. In the following, we will also investigate the reaching condition and the corresponding reaching time for the states to reach the sliding manifold from any initial point.

Theorem 2: For the nominal system with $M = \bar{M}$ and $\tau(t) = \bar{\tau}$ in (8), the states of the closed-loop system implementing the VS controller (14) can reach the sliding manifold from any initial point, and the reaching time is no larger than

$$t_{\max} = \frac{\bar{\tau}^2}{\delta\bar{M}}|k_S x_1(0) + x_2(0)|. \quad (23)$$

Proof: With $M = \bar{M}$ and $\tau(t) = \bar{\tau}$, the nominal system is given as follows:

$$\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_2 &= \frac{\bar{M}}{\bar{\tau}^2} - \left(\frac{\bar{M}}{\bar{\tau}^2} + \frac{(x_2 + C_0)^2}{2\bar{M}}\right)p(t). \quad (24)\end{aligned}$$

Introduce the Lyapunov functional as

$$V(x,t) := \frac{1}{2}S^2(x,t) \quad (25)$$

i.e.,

$$|S(x,t)| = \sqrt{2V}. \quad (26)$$

Find the time derivative of $V(x,t)$ along the trajectories of system (24) with control (14)

$$\begin{aligned}\dot{V} = S\dot{S} &= S\left(k_S x_2 + \frac{\bar{M}}{\bar{\tau}^2} - \bar{b}(x_2)p(x,t)\right)\\ &= S\left[k_S x_2 - \bar{b}(x_2)\left(\alpha\frac{T_1}{\sqrt{2}}k_S|x_2/r(t)| + \delta\right)\operatorname{sign}(S)\right]\\ &\le -|S|\bar{b}(x_2)\left(\alpha\frac{T_1}{\sqrt{2}}k_S|x_2/r(t)| + \delta\right) + k_S|x_2||S|\\ &\le -|S|(\delta\bar{b}(x_2) + \alpha k_S|x_2| - k_S|x_2|)\\ &\le -\frac{\delta\bar{M}}{\bar{\tau}^2}|S|. \quad (27)\end{aligned}$$

Recalling (26) yields

$$\dot{V} \le -\frac{\sqrt{2}\delta\bar{M}}{\bar{\tau}^2}\sqrt{V} := -\rho\sqrt{V} \quad (28)$$

Since the Lyapunov function decays at a finite rate, it vanishes and sliding mode occurs after a finite time interval. Thus, the reaching condition is fulfilled.

Note that the solution to the differential inequality $V(x,t)$ in (28) is nonnegative and is bounded by

$$V(x,t) < \left(-\frac{\rho}{2}t + \sqrt{V_0}\right)^2 \quad V_0 = V(0). \tag{29}$$

From (26) and (29), we come up with an estimate of the reaching time t_{reach}

$$t_{\text{reach}} \le t_{\max} = \frac{2\sqrt{V}}{\rho} = \frac{\bar{\tau}^2}{\delta \bar{M}} |k_S x_1(0) + x_2(0)| \tag{30}$$

and complete the proof. ■

With a view toward implementation of the proposed VS-based AQM scheme (14), we have to take the saturation constraint $0 \le p(t) \le 1$ into consideration, which results in the following modification:

$$p_{\text{real}}(t) = \Psi(p(t)) \tag{31}$$

where

$$\Psi(\xi) := \begin{cases} 0, & \text{if } \xi < 0 \\ \xi, & \text{if } 0 \le \xi \le 1 \\ 1, & \text{if } \xi > 1. \end{cases}$$

Such modification decreases the control magnitude, which may introduce chattering behavior and even system divergence. Note that the small fluctuation of the queue, caused by the chattering of the state trajectory around the sliding manifold, is acceptable in network buffer management.

Remark 2: Theoretically, the VS controller in the form of (14) can achieve robust stability regarding any uncertainty bounds of M and $\tau(t)$ by choosing the control magnitude large enough, which may result in heavy saturation and performance deterioration. This is a tradeoff with respect to the system robustness and performance.

The previous algorithm can be validated using the following example:

Example 1: Consider M homogeneous TCP connections sharing a link with capacity $C_0 = 1250$ packet/s,[1] where we choose M randomly distributed in $[80, 120]$ with $80 = M^- \le M \le M^+ = 120$, and the RTT is time varying obeying

$$0.15 \text{ s} = T_0 \le \tau(t) = 0.2 + 0.05 \sin\left(\frac{2\pi}{100}t\right) \le T_1 = 0.25 \text{ s}.$$

Thus, $\bar{M} = 100$ and $\bar{\tau} = 0.2$. We further assume the desired queue $q_d = 150$ packet. Recalling (16) gives

$$\gamma = \frac{1}{4}\left(\frac{\bar{M}}{M^-} + \frac{\bar{\tau}}{T_0}\right)\left(\frac{M^+ - M^-}{M^+ + M^-} + \frac{T_1 - T_0}{T_1 + T_0}\right) = 0.29.$$

Based on Theorem 1, we choose $k_S = 10, \alpha = 1$, and $\delta = 0.3$. Inserting these parameters into (14), we come up with the VS-based AQM controller. The MATLAB simulations are depicted in Figs. 2 and 3, where the nominal system with $M = \bar{M}$ and $\tau(t) = \bar{\tau}$ is also simulated as comparison.

[1]corresponds to a 10 Mb/s link with average packet size 1000 Bytes.

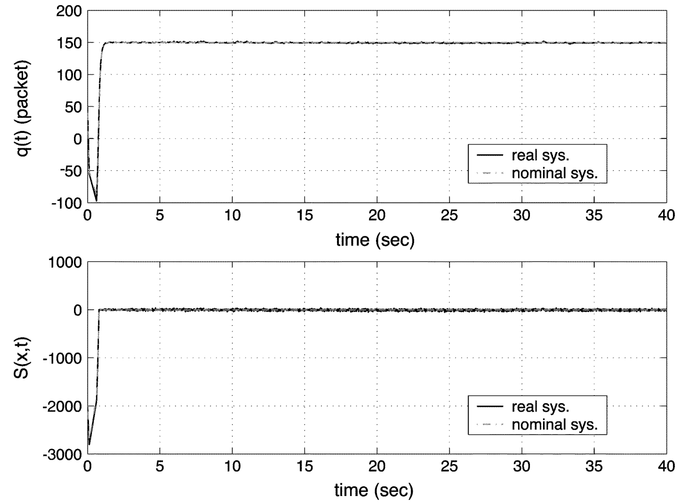

Fig. 2. System responses using the VS controller.

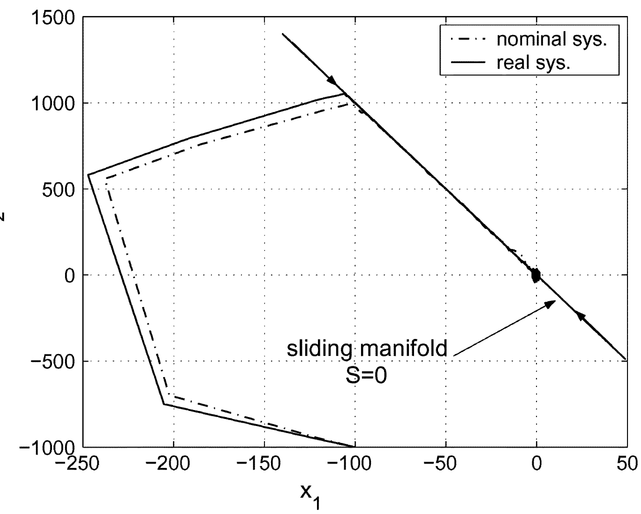

Fig. 3. Phase portrait of the closed-loop system.

IV. Related Work

Current queue management schemes, such as DropTail and RED, have been observed to have some drawbacks such as link underutilization, heavy queue fluctuation and global synchronization. Several improved AQM schemes have been proposed to remedy these drawbacks based on more systematic analysis of the TCP dynamics [1], [2], [11], [14], [16], [19]. In what follows, we provide a brief review and comparison of some representative TCP/AQM algorithms in terms of their motivations, performance objectives and methodologies.

1) **Modifications and alternatives of RED**. To remedy RED's drawbacks (e.g., parameter sensitivity), several approaches have been proposed in the recent years, e.g., FRED [16], balanced RED (BRED) [1] and SRED [19]. FRED and BRED share the same objective of improving per-flow fairness by monitoring the per-flow queue length qlen_i and tuning the dropping rate correspondingly. Meanwhile, BRED can be considered as an extension of FRED in the sense that BRED has a refined

dropping policy by dividing the space of qlen_i into four regions with four dropping probabilities in each of the regions. Note that both of them use only the queue length as the congestion index to calculate dropping probability $p(t)$, which is believed to be difficult to achieve high link utilization and low packet loss ratio simultaneously [2]. On the other hand, SRED was proposed to alleviate the heavy fluctuation of the instantaneous queue caused by RED. Using the estimate of the number of active flows N, SRED modified the dropping probability as an increasing function of N. Thus, SRED can adaptively adjust its dropping probability with respect to TCP flow number. Although the previous modifications outperform RED in the aspects of fairness and throughput, they are essentially heuristic without systematic analysis.

2) **AQM schemes based on equilibrium structure and utility optimization**. The congestion control system can be considered as a convex optimization problem for a certain aggregate utility function, where TCP/AQM can be interpreted as carrying out a gradient algorithm to maximize aggregate source utility. Schemes in this category (e.g., REM [2] and AVQ [14]) aim at achieving both high utilization and low packet loss ratio by introducing the price function as congestion index.

In REM (Random Exponential Marking), a price function $\mu(k)$ is defined based on the queuing information and the incoming rate

$$\mu(k+1) = \max(0, \mu(k)+\gamma(\alpha(q(k)-q_{\mathrm{ref}})+x(k)-C)) \quad (32)$$

where γ and α are constants, $q(k)$ and $x(k)$ are the queue length and the incoming rate, respectively, and C is the link capacity. Correspondingly, REM uses an adaptive marking probability $p(k)$ to regulate the queue length

$$p(k+1) = 1 - \phi^{-\mu(k)} \quad (33)$$

where $\phi > 1$ is a constant. Note that the term $\alpha(q(k) - q_{\mathrm{ref}}) + x(k) - C$ in (32) is in fact the weighted mismatch of the queue and the rate, which is similar to the sliding manifold $S(x,t)$ in our design.

The AVQ (Adaptive Virtual Queue) is another adaptive AQM scheme, whose marking probability is determined by the virtual queue capacity updated according to

$$\dot{\tilde{C}} = \alpha(\gamma C - x(k)) \quad (34)$$

where γ is the desired link utilization. Based on the linearized TCP dynamics, AVQ calculates marking probability using only the incoming rate $x(k)$, so it is primarily based on incoming traffic rate to provide early congestion feedback.

Scalable control scheme proposed in [17] uses link's price $p_l(t)$ as congestion index and marks packets with probability $1 - \phi^{-p_l(t)}, \phi > 1$. The link updates its price $p_l(t)$ using the aggregate input rate $y_l(t)$ according to

$$\dot{p}_l(t) = \begin{cases} \frac{y_l - c_l}{c_l}, & \text{if } p_l(t) > 0 \\ \max\left\{0, \frac{y_l - c_l}{c_l}\right\}, & \text{if } p_l(t) = 0 \end{cases} \quad (35)$$

in which c_l is the *virtual capacity* that is strictly less than the real link capacity. The source will set its sending rate as an exponential function with respect to the aggregate price $q_i(t)$, i.e., $x_i(t) = x_{\max,i} e^{-(\alpha_i q_i(t))/(M_i \tau_i)}$. To utilized this scheme, the current TCP's congestion control and avoidance scheme has to be changed which is not an easy task.

3) **AQM schemes based on feedback control theory**. The dynamic models of TCP make it possible to design AQM in the literature of feedback control theory. In [9], RED was considered as a AQM controller whose parameters can be analyzed based on a linearized TCP model. An extension to this work is PI AQM [11], whose marking probability is updated based on the queue length as

$$p(k+1) = p(k) + a(q(k+1) - q_{\mathrm{ref}}) - b(q(k) - q_{\mathrm{ref}}) \quad (36)$$

where a and b are constants.

It has been shown in [11] that the PI AQM scheme can outperform RED in terms of system response and steady-state error. On the other hand, the PI controller has some inherent limitations: 1) the linearization inevitably introduces model error; 2) it is based on the frequency domain analysis which is invalid for time-varying systems (TCP dynamics are essentially time varying); 3) although gain-phase margin can be analyzed for the PI controller, it can not incorporate robustness directly in the design. We also notice that a robust AQM [21] was recently proposed based on infinite dimensional $\mathcal{H}^\infty$ optimization. It allows for the uncertainties of the RTT, the active TCP sessions and the available link capacity.

It is worth mentioning that DropTail can be viewed as a variable structure controller in the sense that its dropping probability is given as

$$p(t) = \begin{cases} 1, & \text{if } q(t) - q_{\max} \geq 0 \\ 0, & \text{if } q(t) - q_{\max} < 0 \end{cases} \quad (37)$$

which is similar to the typical structure of sliding mode control. From the point of view of VS control, the reason why DropTail performs poorly is that it could not enforce a stable sliding manifold.

The idea of sliding mode control was also discussed in [23], where a sliding mode AQM algorithm (which is called SMVS) was proposed for the linearized delay-free system. The dropping rate of SMVS is calculated by

$$p(t) = \begin{cases} \alpha(q(t) - q_d), & \text{if } (q(t) - q_d)\sigma > 0 \\ \beta(q(t) - q_d), & \text{if } (q(t) - q_d)\sigma < 0 \end{cases} \quad (38)$$

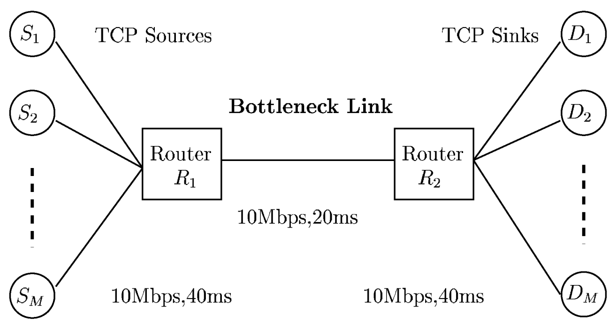

Fig. 4. Dumbbell network topology for *ns* simulations.

where

$$\sigma = w(q(t) - q_d) + \frac{d}{dt}(q(t) - q_d) \quad w > 0 \tag{39}$$

is the sliding manifold and α, β are constants. It has been shown in [23] that SMVS has better robustness and performance than PI AQM. However, the SMVS controller has a fundamental mistake. Taking the constraint $0 \leq p(t) \leq 1$ into consideration, (38) is valid only in the range of $|q(t) - q_d| \leq 1/\max\{|\alpha|, |\beta|\}$. Using the recommended values $\alpha = -\beta = 0.96$ in the paper, we find that SMVS will always work at the saturation states of $p = 0$ or $p = 1$, except when $|q(t) - q_d| \leq 1/0.96 = 1.04$, indicating that it is degraded to a DropTail scheme at most of the time. Furthermore, the theoretical analysis in [23] is invalid for the implementation of SMVS [23, Eq. 10], so that the stability of the AQM algorithm can not be guaranteed.

Compared with the previous schemes, the VS-based AQM proposed in the present paper is the only one that is directly designed for the time-varying nonlinear TCP dynamics, which has good robustness and fast response inherited from sliding mode control. More comprehensive comparison will be given in Section V using *ns-2* simulations.

V. Simulation Results

To validate the performance and the robustness of the proposed VS AQM, we implement it in *ns-2* and conduct a simulation study in different scenarios. Some representative AQM schemes, namely, DropTail, RED [8], REM [2] and PI [11], are also simulated for the purpose of comparison. Note that the sliding mode AQM (SMVS) in [23] is not considered for comparison because the simulations in [23] are questionable due to the severe problem discussed in the previous section.

A. Simulation Configuration

The dynamic behaviors of the previous AQM schemes are simulated under a variety of network topologies and traffic sources. In particular,we consider the dumbbell network topology depicted in Fig. 4, where M TCP connections share a single bottleneck link. We assume that the TCP sources always have data to send. The links between the TCP sources and the router R_1 are 10 Mbps links with a 40 ms propagation delay, which are the same as those between the TCP sinks and the outer R_2. Router R_1 is connected to R_2 through a 10 Mbps 20 ms delay link. The maximum buffer size of each router is set to 300 packets (of size 1000 bytes). Meanwhile, we consider the network topology with multiple bottleneck links (Fig. 5), where the maximum buffer of each router is 200 packets, the bandwidth and the propagation delay of each link are indicated in Fig. 5 and each sender-receiver pair has n_{cross} TCP connections as cross traffic. In both scenarios, TCP-Reno is used as the transport agent.

The parameters used in VS-based AQM (14) are: $\bar{M} = 100, \bar{\tau} = 0.2, K_s = 10, \alpha = 1$, and $\delta = 0.3$ (see example 1 for details). In PI controller (36), we use the suggested parameter values $a = 1.822 \times 10^{-5}$ and $b = 1.816 \times 10^{-5}$ given in [10]. The desired queue length is set to $q_d = 150$ packets for VS control and PI control. The parameters of RED are set as recommended in http://www.aciri.org/floyd/REDparameters.txt. For REM (32) and (33), the parameters are set as $\alpha = 0.1, \gamma = 0.001$, and $\phi = 1.001$, which are recommended in [2]. In the following simulations, ECN is enabled for VS AQM, PI, REM, and RED, respectively, where $p(t)$ corresponds to the marking rate, and packet loss is observed only when the buffer overflows.

B. Scenario of Single Bottleneck Topology

1) Performance Comparison of Different AQM Schemes: In this experiment, we choose $M = 100$ in Fig. 4, which corresponds to 100 greedy FTP flows sharing the bottleneck link. The system response using the VS controller is depicted in Fig. 6, where the performance shows fast response and the stabilized queue size. Meanwhile, we repeat the same experiment using DropTail, RED, PI, and REM, respectively, and depict their instantaneous queues in Fig. 7.

Note that the heavy oscillation of DropTail [Fig. 7(a)] coincides with our analysis in Section IV from the point of view of sliding mode control. As compared to the AQM schemes shown in Fig. 7, we clearly see that the VS AQM outperforms other schemes in terms of system stability and performance, which implies higher link utilization, lower packet loss ratio and smaller queue fluctuation. Although the PI controller is also a control theory-based design which could regulate the queue to the desired value, its transient response is sluggish, which deteriorates its performance [(Fig. 7(c)].

2) Performance Under Dynamic Traffic Changes: In this scenario, we provide some time-varying dynamics and investigate the performance of the VS controller and other representative schemes. We use 150 TCP connections at time $t = 0$. At time $t = 40, 50$ of the TCP connections stop transmitting data, and at time $t = 70$ they resume transmitting again. The queue evolution is depicted in Fig. 8. Note that PI [Fig. 8(b)] and REM [Fig. 8(c)] are not very robust with respect to such connection number variation, which result in heavy queue fluctuation during 40–70 s. Although RED [Fig. 8(a)] is not very sensitive in this scenario, it tends to over-mark the incoming traffic so that the link utilization is degraded. As is evident from [Fig. 8(d)], the VS controller is very robust against the variation of connections and keeps very good response even in the presence of such variations.

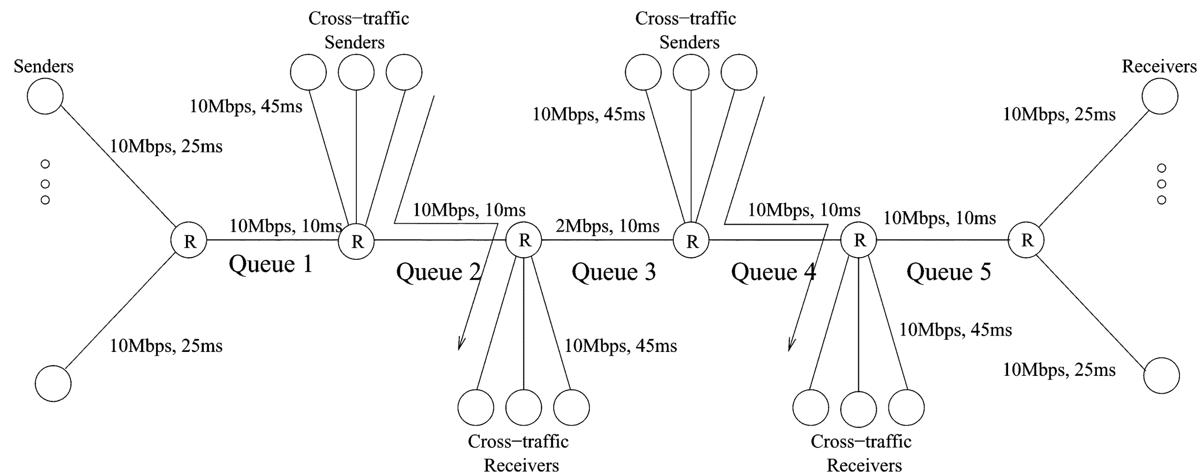

Fig. 5. Network topology with multiple bottleneck links.

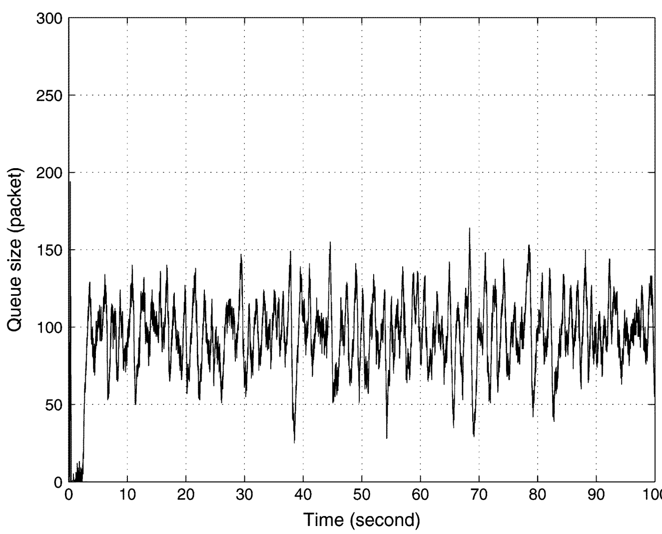

Fig. 6. Instantaneous queue size using VS control.

3) Robustness w.r.t. Number of TCP Connections: The performance and robustness of the VS scheme are explored with respect to different TCP loads. We conduct simulations with the same setting as in Experiment 1, except that the number of connections varies from 50 to 250. Fig. 9 plots the average queue length (from 20 to 100 s) for different AQM schemes. Correspondingly the link utilization and the packet loss ratio are depicted in Figs. 10 and 11, respectively.

It is observed that the VS controller can robustly stabilize the queue length around 100 packets. The average queue lengths of REM and RED vary slightly with respect to the flow number, while the average queue of PI blows up when the flow number increases, which is due to the inherent nature of PID control that system response is highly dependent on the system parameters.

From Fig. 10, we clearly see that the VS and PI controller have better link utilization than REM and RED, and the link utilization of REM and RED are sensitive to the variation of the flow numbers. Note that the packet loss ratio of PI increases with respect to higher loads (Fig. 11), which is not desirable for congestion control. REM has lower drop rate compared with PI, but its link utilization is not that good. Notice that in Fig. 11, the drop rate for RED is not plotted due to the fact that RED has a much higher drop rate in the range between 6% and 13% in this simulation. Compared with RED, REM, and PI, the VS controller has much better performance in terms of robustly stabilized queue length, high link utilization and low packet loss ratio (in this experiment, the packet loss ratio for VS AQM is always 0).

The previous analysis shows that the control theory-based AQM designs, say, PI, and VS AQM, have better link utilization by stabilizing the queue length to the desired values. Note that the system response (response time and overshoot) of PI is deteriorated by higher loads. Thus, the high link utilization of PI is in the expense of high packet drop rate when the number of TCP flows increases. On the other hand, the VS controller keeps fast response, stability robustness with respect to a large range of variation of TCP flow numbers, which is well-suited as an AQM scheme.

4) Comparison with PI AQM Under Higher TCP Loads: It has been shown in [10] that the PI controller has robustness with respect to the number of connections, although a larger number of connections results in a slower system response. In this experiment, we set the flow number to $M = 300$ and compare the system response of PI and the VS AQM. As shown in Fig. 12, the VS controller has very good transient response and very low overshoot. On the other hand, the PI controller exhibits much slower response and larger overshoot, which implies higher packet loss ratio and larger RTT.

5) Performance in the Presence of Short-Lived TCP Flows: In this experiment, we investigate the system performance in the presence of short-lived TCP flows. The single bottleneck topology we considered is depicted in Fig. 4, where 120 greedy FTP flows and 50 short-lived TCP flows share the link. Each short-lived TCP flow is configured to randomly turn ON–OFF in each 5-s period. As shown in Fig. 13, VS AQM has good robustness against the disturbance of such short-lived TCP flows, and can significantly improve the system performance.

6) Robustness w.r.t. RTT: As discussed in Section III, the VS controller has very good robustness against the uncertainty of RTT, which is essential as an AQM scheme. In this experiment, we change the propagation delays in the network topology

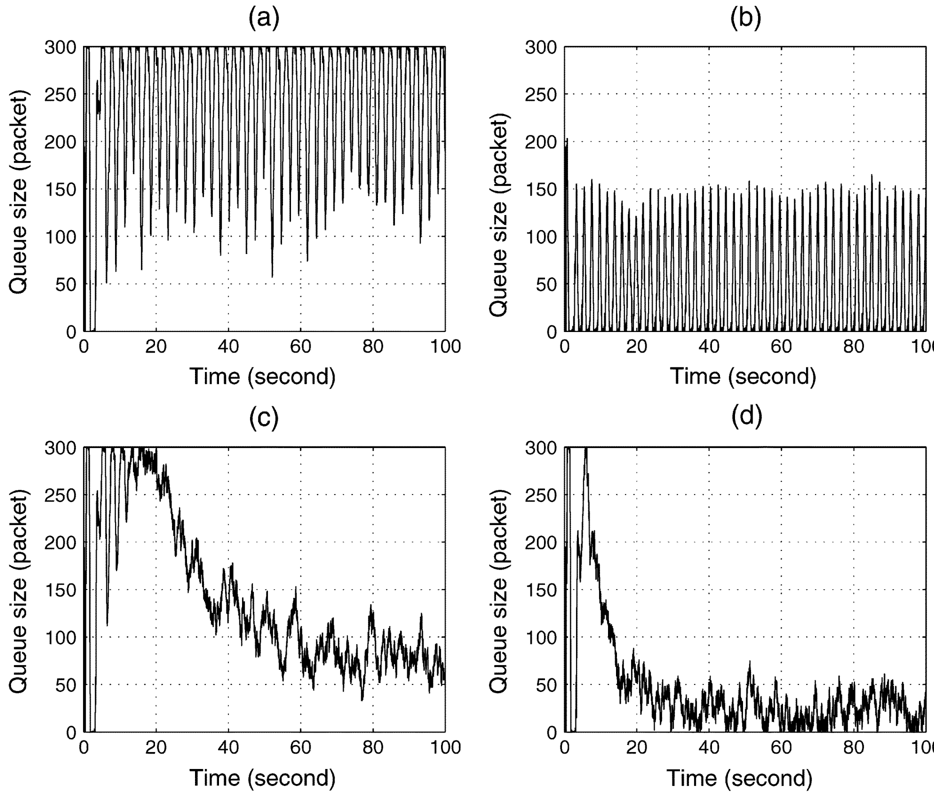

Fig. 7. System responses for DropTail, RED, PI, and REM.

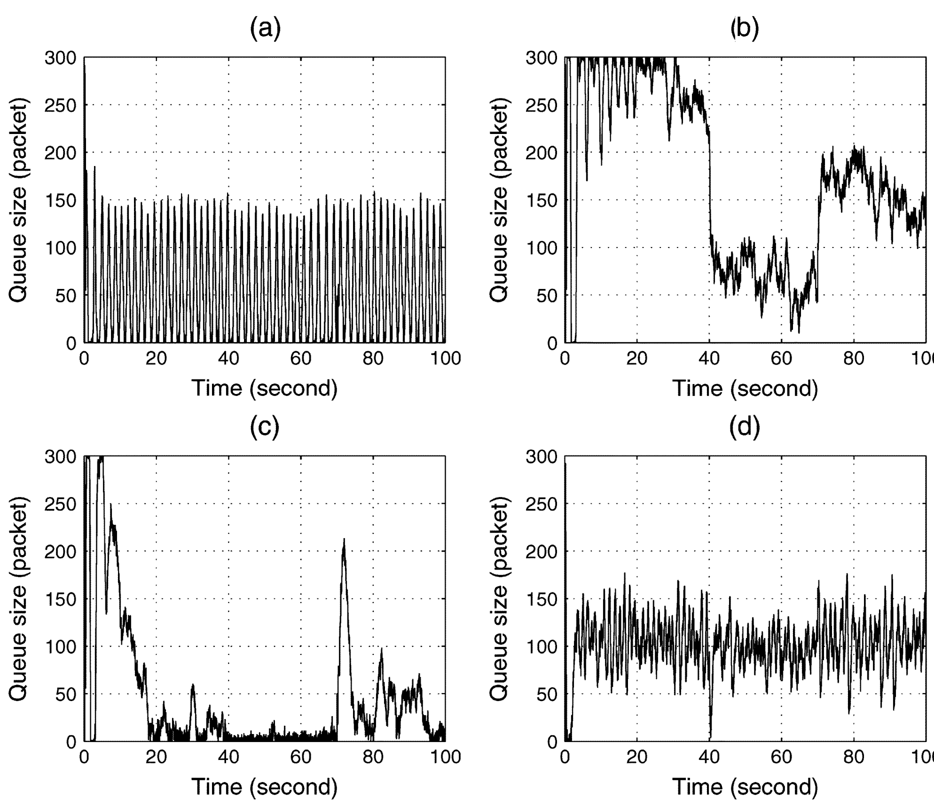

Fig. 8. Queue evolution using RED, PI, REM, and VS control.

(Fig. 4) and evaluate the robustness of the VS controller. First, we set the propagation delay between router R_1 and R_2 to 10 ms and the delays between the routers and the end hosts 2 ms, which corresponds to a much smaller RTT than the nominal one. The regulated queues are depicted in Fig. 14 where the VS controller is still capable of robustly stabilizing the queue length.

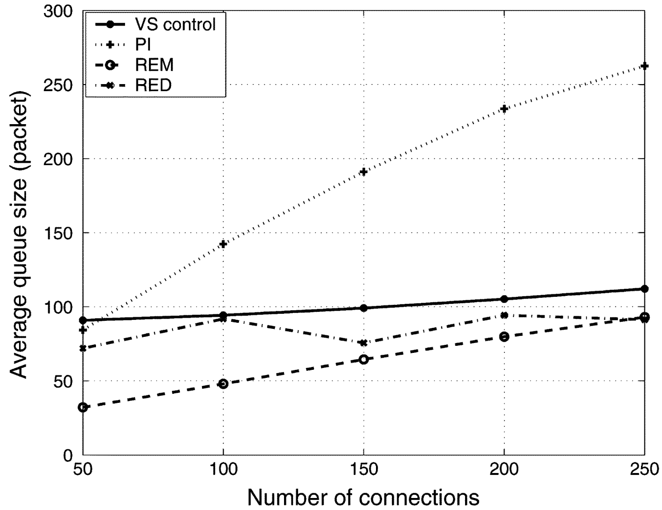

Fig. 9. Average queue length w.r.t. the number of TCP flows.

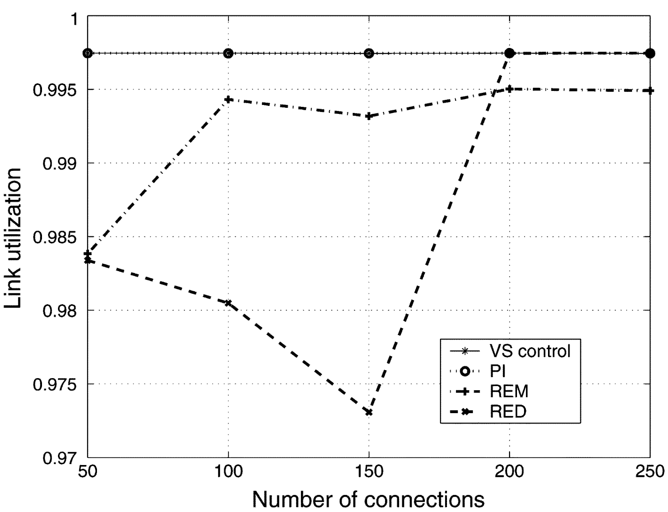

Fig. 10. Link utilization w.r.t. the number of TCP flows.

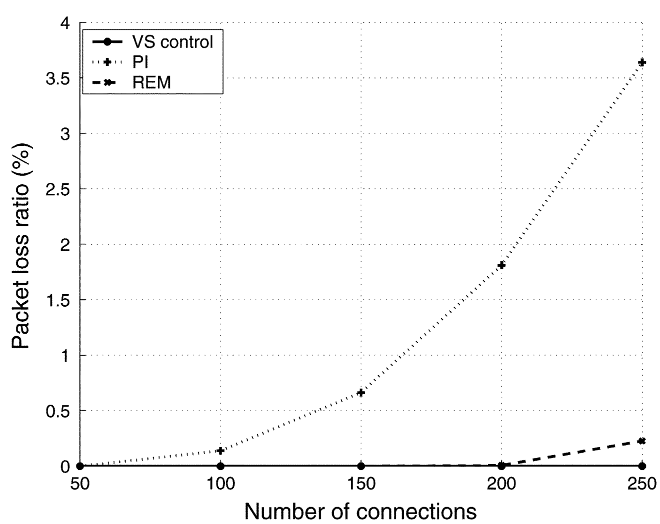

Fig. 11. Packet loss ratio w.r.t. the number of TCP flows.

Note that for this scenario, RED [Fig. 14(a)] also exhibits good performance. Meanwhile, we also consider the scenario with a much larger RTT, where we set the propagation delay between

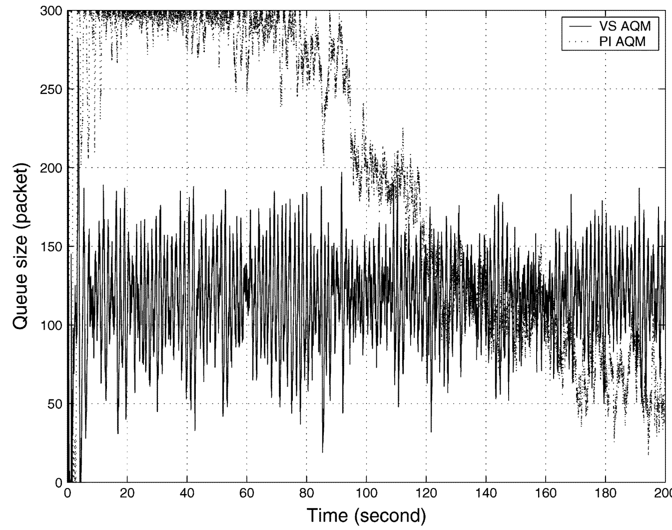

Fig. 12. Comparison of PI and the VS controller.

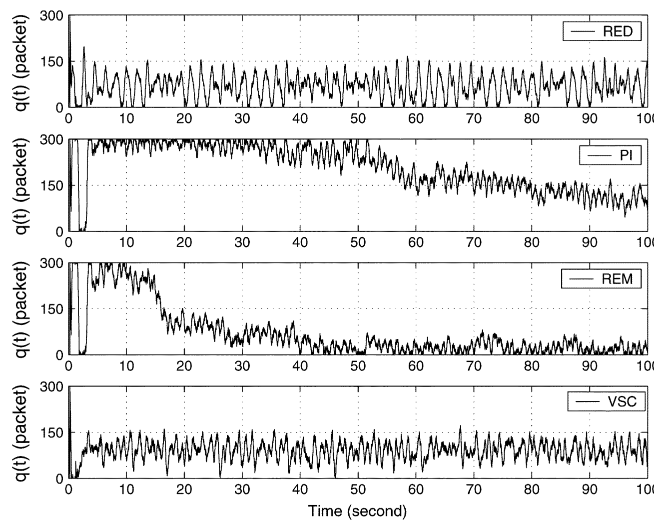

Fig. 13. Performance in the presence of short-lived TCP flows.

R_1 and R_2 to 120 ms and the delays between the routers and the end hosts 20 ms. We repeat the simulation with the flow number $M = 100$ and obtain the system responses depicted in Fig. 15. As we can observe, the VS controller continues to exhibit good performance in the sense of queue stability and fast response, which outperforms other peer schemes.

C. Scenario of Multiple Bottleneck Topology

Using the multiple bottleneck network topology depicted in Fig. 5, we study the behaviors of different AQM schemes in the presence of cross traffic. We set 300 TCP connections with sender at the left hand side and receivers at the right hand side, with 50 TCP flows $(n_{\mathrm{cross}} = 50)$ for each cross traffic sender-receiver pair. The instantaneous queues of Queue 3 for different AQMs are depicted in Fig. 16, and those of Queue 4 are plotted in Fig. 17. Note that Queue 2, 3, and 4 exhibit similar trends. Queue 1 and Queue 5 are almost empty, indicating that these two links are not bottleneck links. Similar results to Figs. 16 and 17 can be obtained under different TCP loads and different cross traffic loads.

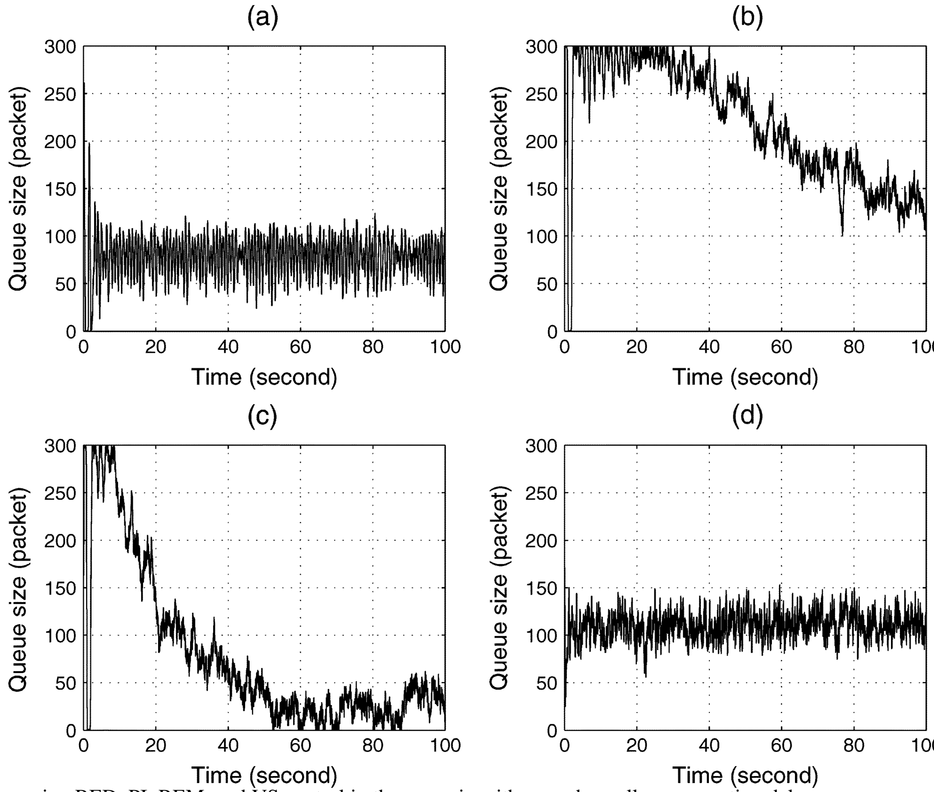

Fig. 14. Queue evolution using RED, PI, REM, and VS control in the scenario with a much smaller propagation delay.

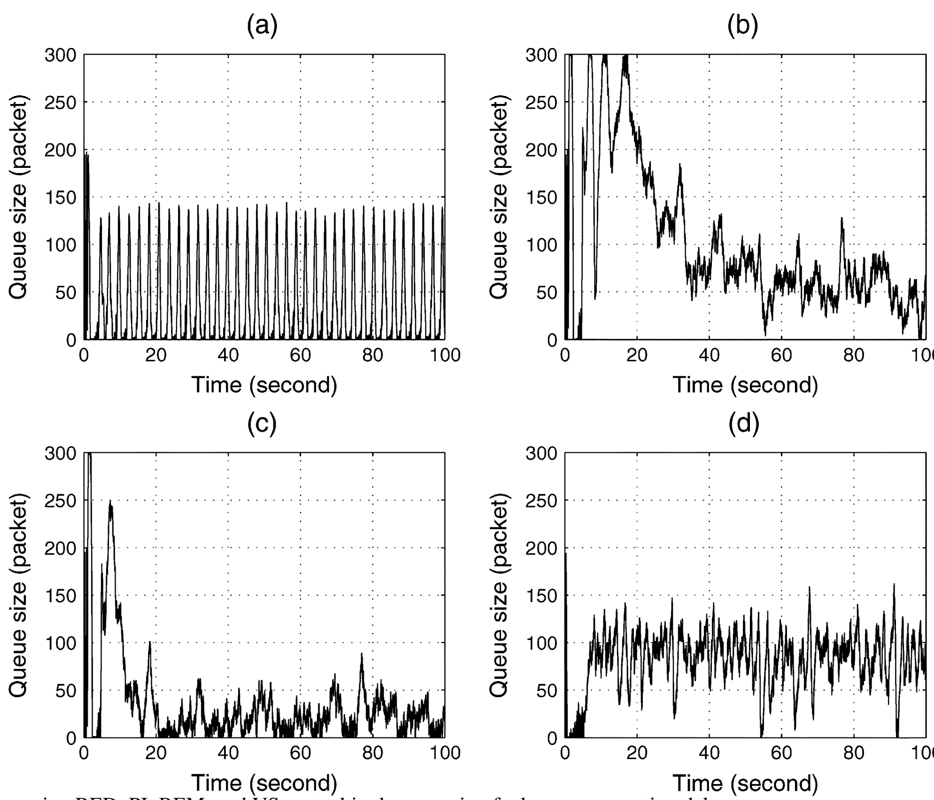

Fig. 15. Queue evolution using RED, PI, REM, and VS control in the scenario of a larger propagation delay.

Once again, the VS controller shows much better performance than other AQM methods. In fact, the performance of RED in this experiment is sensitive to the network configurations (e.g., TCP loads, cross traffic and propagation delays), which affirms coincidence to [3]. On the other hand, REM tends to mark too many packets and keeps too small a queue size, so that the link utilization is lower than other AQMs. Meanwhile, PI AQM suffers a sluggish transient behavior and makes the queue full in its transient period, which results in buffer overflow and packet losses.

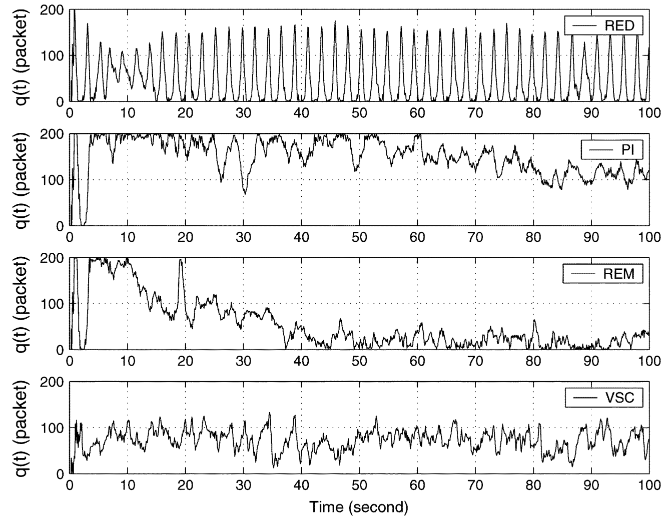

Fig. 16. Evolution of Queue 3 (in packet) using RED, PI, REM, and VS control.

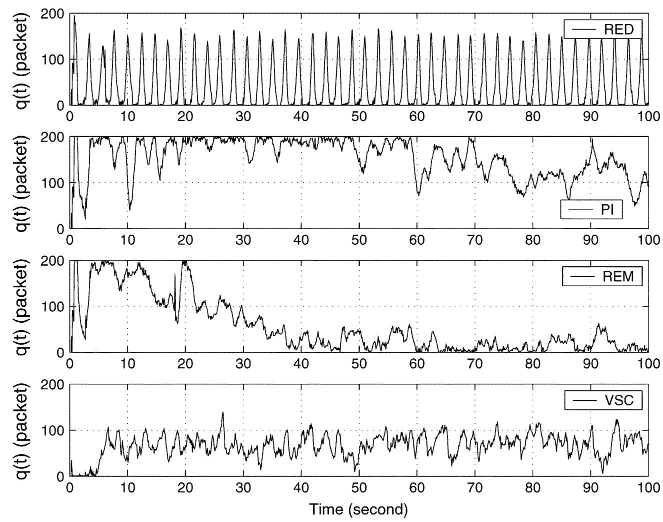

Fig. 17. Evolution of Queue 4 (in packet) using RED, PI, REM, and VS control.

Remark 3: Note that the VS controller has a steady-state error ($\lim_{t\to\infty} q(t) - q_d$) of about 50 packets in the previous scenarios. In fact, the VS control is designed for the delay-free nonlinear system (8) and the corresponding sliding manifold (9) is also delay-free. A more accurate model of the real network traffic is the delayed version of (8)

$$\begin{aligned}\dot{r}(t) =& \frac{Mr(t-\bar{\tau})}{\tau^2(t)r(t)} - \left(\frac{Mr(t-\bar{\tau})}{\tau^2(t)r(t)} + \frac{r(t)r(t-\bar{\tau})}{2M}\right)p(t-\bar{\tau})\\ \dot{q}(t) =& r(t) - C_0, \end{aligned} \tag{40}$$

which was proposed in [13]. The presence of time delays results in $S(x,t) \neq 0$ in the system steady state, which causes the steady-state error of the queue size.

VI. Concluding Remarks

In this paper, we developed a variable structure-based AQM control scheme supporting ECN. We presented guidelines for designing the robust VS sliding mode controller directly for the nonlinear TCP dynamics. The robustness with respect to the RTTs and the number of the active TCP sessions was analyzed, and the asymptotic properties of the closed-loop system was discussed. It was shown that the VS controller, from control theoretic point of view, has many desirable properties such as good robustness and fast system response. We also provided *ns* simulations in different scenarios to validate our results. The simulation experiments showed that the proposed AQM scheme performs better than a number of well-known AQM schemes in terms of packet loss ratio, link utilization and queue fluctuation. A challenging extension of this work is to consider VS control for the TCP model in the presence of time delays as shown in (40), which we are currently investigating.

References

[1] F. M. Anjum and L. Tassiulas, "Fair bandwidth sharing among adaptive and nonadaptive flows in the internet," in *Proc. IEEE INFOCOM*, New York, Mar. 1999, pp. 1412–1420.

[2] S. Athuraliya, S. H. Low, V. H. Li, and Q. Yin, "REM: Active queue management," *IEEE Network*, vol. 15, no. 3, pp. 48–53, May-Jun. 2001.

[3] M. Christiansen, K. Jeffay, D. Ott, and F. D. Smith, "Tuning RED for web traffic," in *Proc. ACM SIGCOMM*, Stockholm, Sweden, Sep. 2000, pp. 139–150.

[4] The network simulator ns-2 (2000). [Online]. Available: http://www.isi.edu/nsnam/ns/

[5] B. Braden *et al.*, "Recommendations on queue management and congestion avoidance in the Internet," Internet Engineering Task Force, RFC-2309, Apr. 1998.

[6] W. Feng, D. Kandlur, and K. Shin, "Stochastic fair blue: A queue management algorithm for enforcing fairness," in *Proc. IEEE INFOCOM*, Alaska, Apr. 2001, pp. 1520–1529.

[7] V. Firoiu and M. Borden, "A study of active queue management for congestion control," in *Proc. IEEE INFOCOM*, Tel-Aviv, Israel, Mar. 2000, pp. 1435–1444.

[8] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Netw.*, vol. 1, no. 4, pp. 397–413, Aug. 1993.

[9] C. Hollot, V. Misra, D. Towsley, and W. Gong, "A control theoretic analysis of RED," in *Proc. IEEE INFOCOM*, Alaska, Apr. 2001, pp. 1510–1519.

[10] ——, "On designing improved controllers for AQM routers supporting TCP flows," in *Proc. IEEE INFOCOM*, Alaska, Apr. 2001, pp. 1726–1734.

[11] ——, "Analysis and design of controllers for AQM routers supporting TCP flows," *IEEE Trans. Autom. Control*, vol. 47, no. 6, pp. 945–959, Jun. 2002.

[12] V. Jacobson and M. Karels, "Congestion avoidance and control," in *Proc. ACM SIGCOMM*, CA, Aug. 1988, pp. 314–329.

[13] F. Kelly, *Mathematics Unlimited—2001 and Beyond*, B. Engquist and W. Schmid, Eds. Berlin, Germany: Springer-Verlag, 2001. Mathematical modeling of the Internet.

[14] S. Kunniyur and R. Srikant, "Analysis and design of an adaptive virtual queue (AVQ) algorithm for active queue management," in *Proc. ACM SIGCOMM*, San Diego, CA, Aug. 2001, pp. 123–134.

[15] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of ethernet traffic (extended version)," *IEEE/ACM Trans. Netw.*, vol. 2, no. 1, pp. 1–15, Feb. 1994.

[16] D. Lin and R. Morris, "Dynamics of random early detection," in *Proc. ACM SIGCOM*, Sep. 1997, pp. 127–137.

[17] S. Low, F. Paganini, J. Wang, S. Adlakha, and J. Doyle, "Dynamics of TCP/RED and a scalable control," in *Proc. IEEE INFOCOM*, New York, Jun. 2002, pp. 239–248.

[18] V. Misra, W. Gong, and D. Towsley, "A fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED," in *Proc. ACM SIGCOMM*, Stockholm, Sweden, Sep. 2000, pp. 151–160.

[19] T. J. Ott, T. V. Lakshman, and L. H. Wong, "SRED: Stabilized RED," in *Proc. IEEE INFOCOM*, New York, Mar. 1999, pp. 1346–1355.

[20] F. Paganini, "A global stability result in network flow control," *Syst. Control Lett.*, vol. 46, no. 3, pp. 165–172, Jul. 2002.

[21] P.-F. Quet and H. Özbay, "On the robust controller design for active queue management scheme supporting TCP flows," *IEEE Trans. Autom. Control* , vol. 49, no. 6, pp. 1031–1036, Jun. 2004.

[22] K. K. Ramakrishnan, S. Floyd, and D. Black, The Addition of Explicit Congestion Notification (ECN) to IP, Internet Engineering Task Force, RFC-3168, Sep. 2001.

[23] F. Ren, X. Ying, Y. Ren, and X. Shan, "A robust active queue management algorithm based on sliding mode variable structure control," in *Proc. IEEE INFOCOM*, New York, Jun. 2002, pp. 13–20.

[24] V. Utkin, *Sliding Modes in Control and Optimization*. Berlin, Germany: Spring-Verlag, 1992.

[25] V. Utkin, J. Guldner, and J. Shi, *Sliding Mode Control in Electromechanical Systems*. New York: Taylor & Francis, 1999.

Peng Yan (S'02–M'03) received the B.S. and M.S. degrees in electrical engineering from Southeast University, Nanjing, P.R. China, in 1997 and 1999, respectively, and the Ph.D. degree from the Department of Electrical Engineering, The Ohio State University, Columbus, in 2003.

He is currently working as a Postdoctoral Researcher for the Collaborative Center of Control Science, The Ohio State University. His research interests include robust control, time delay systems, internet congestion control and active queue management, aerodynamics modeling and cavity flow control, and real-time control systems.

Yuan Gao received the B.S. and M.S. degrees in electrical engineering from Tsinghua University, Beijing, P.R. China, in 1994 and 1997, respectively, and the Ph.D. degree from the Department of Electrical Engineering, The Ohio State University, Columbus, in 2002.

He is currently working as a Member of Technical Staff at Bell Laboratories, Lucent Technologies, Holmdel, NJ. His current research interests include internet congestion control, active queue management, TCP enhancement, multicast routing, and high-speed network monitoring and measurement.

Hitay Özbay (S'89–M'89–SM'97) received the B.S. degree in electrical engineering from Middle East Technical University, Ankara, Turkey, in 1985, the M.Eng. degree in electrical engineering from McGill University, Montreal, Canada, in 1987, and the Ph.D. degree in control sciences and dynamical systems from the University of Minnesota, Minneapolis, in 1989.

He was an Assistant Professor of Electrical Engineering at the University of Rhode Island from 1989 to 1990. Since 1991, he has been with The Ohio State University, Columbus, where he is a Professor of Electrical and Computer Engineering. Currently, he is also a Professor of Electrical and Electronics Engineering at Bilkent University, Ankara, Turkey.